

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-141208

(43)Date of publication of application : 02.06.1995

(51)Int.Cl.

G06F 9/46
G06F 9/46

(21)Application number : 05-308640

(71)Applicant : SONY CORP

(22)Date of filing : 15.11.1993

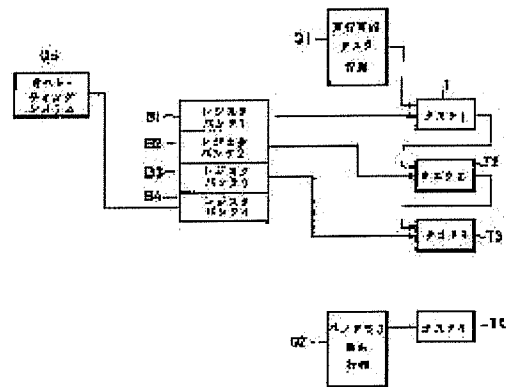
(72)Inventor : FUJISHIRO TOSHIHIKO
OIWA JUNJI

(54) MULTITASK PROCESSOR

(57)Abstract:

PURPOSE: To shorten the dispatch time while the number of tasks which can be executed in parallel is not limited by providing plural register banks which are occupied corresponding to respective tasks and dynamically allocating the register banks to the tasks.

CONSTITUTION: The tasks T1-T4 are made queued in an executable task matrix Q1. When the tasks T1 and T2 among those tasks T1 and T4 are started in this order, a register bank B1 is allocated as a resource to the task T1 and a register bank B2 is allocated as a resource to the task T2. Then, when the tasks T3 and T4 are started, a register bank B3 is free, so the register bank B3 is allocated as a resource to the task T3. Here, the task T4 is queued in the free bank queue Q2. Then, an operating system OS allocates the task T4, queued in the free bank queue Q2, to a register bank of a task which completes its process.



LEGAL STATUS

[Date of request for examination] 13.11.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3644042

[Date of registration] 10.02.2005

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平7-141208

(43)公開日 平成7年(1995)6月2日

(51)Int.Cl.⁵

G 0 6 F 9/46

識別記号

3 4 0 B 8120-5B

3 1 3 C 8120-5B

片内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数2 F D (全 5 頁)

(21)出願番号 特願平5-308640

(22)出願日 平成5年(1993)11月15日

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 藤代 俊彦

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72)発明者 大岩 純二

東京都品川区北品川6丁目7番35号 ソニー株式会社内

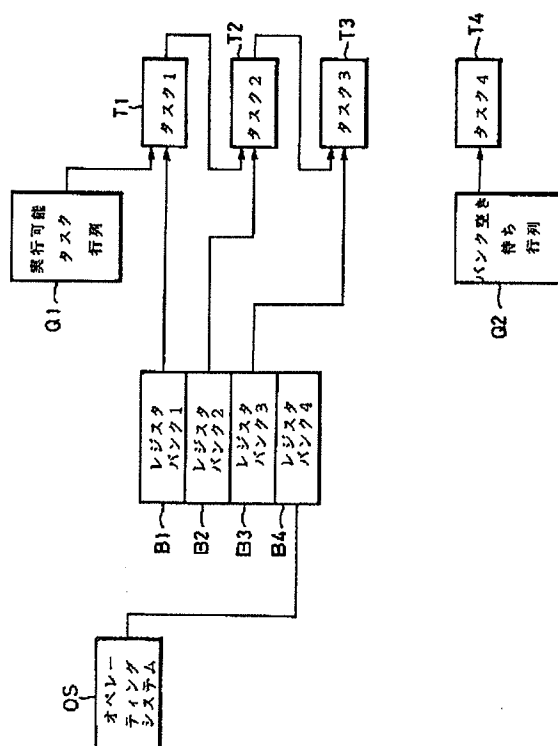
(74)代理人 弁理士 杉浦 正知

(54)【発明の名称】 マルチタスク処理装置

(57)【要約】

【目的】 リアルタイムオペレーティングシステムを用いたマルチタスク処理装置で、並行して実行できるタスクの数が制限されず、ディスパッチ時間を短縮できるようにする。

【構成】 各タスクT1～T4に対応して占有される複数のレジスタバンクB1～B4を設ける。このため、コンテキスト等の退避／復帰がレジスタバンクB1～B4を切り替えることにより行なえ、ディスパッチ時間が短縮される。この複数のレジスタバンクB1～B4を各タスクに動的に割り当てることにより、実行されるタスクの数がレジスタバンクの数に制限されなくなる。



【特許請求の範囲】

【請求項1】 複数のタスクが並列的に実行されるマルチタスク処理を行うマルチタスク処理装置において、各タスクに対応して占有される複数のレジスタバンクを有し、

上記複数のレジスタバンクが各タスクに動的に割り当てられるようにしたマルチタスク処理装置。

【請求項2】 更に、レジスタバンクの空き待ち行列手段を有し、

上記各レジスタバンクは、そのバンクのタスクの処理が終了した時点でそのタスクを解放し、タスクを解放したバンクに上記空き待ち行列手段に待機されているタスクを割り当てるようにした請求項1記載のマルチタスク処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は、リアルタイムオペレーティングシステムを用いたマルチタスク処理装置に関する。

【0002】

【従来の技術】リアルタイムオペレーティングシステムは、例えば組み込み機器用のオペレーティングシステムとして用いられる。リアルタイムオペレーティングシステムは、一定時間以内での応答を保証するようなオペレーティングシステムであり、複数のタスクを並列的に実行させるためのマルチタスク機能が不可欠である。

【0003】リアルタイムオペレーティングシステムは、タスク管理機能、タスク付属同期機能、同期・通信機能、メモリプール管理機能、割り込み管理機能、時間管理機能、システム管理機能を有している。

【0004】タスク管理機能は、タスクの状態を直接操作したり、参照したりする機能である。タスク付属同期機能は、タスクをサスペンド状態にして一時的に実行を停止する機能とそれを解除する機能等を含むタスクに予め付加されている同期機能である。同期・通信機能は、セマフォ、イベントフラグ、メールボックスを用いたタスクとは独立した同期機能及び通信機能である。メモリプール管理機能は、カーネルによってメモリブロックの割当及びメモリプールの管理を行う機能である。割り込み管理機能は、外部割り込みの禁止/許可及び外部割り込みに対するハンドラの定義を行う機能である。時間管理機能は、システムクロックの設定や参照する機能、タスクを遅延させる機能及び指定した時刻に起動されるハンドラの定義を行う機能である。システム管理機能は、カーネルのバージョンを参照する機能である。

【0005】オペレーティングシステムがタスクを管理するための制御領域は、TCB (Task Control Block) と呼ばれている。TCBは、各タスクを管理するために、タスク毎に必要な。TCBには、タスクに割当られたID番号であるタスクID、タスクの状態を示す

タスクステータス、タスク起動時のタスクの優先度を示すタスク優先度、タスクに割り当てられたスタックポインタを示すスタックポインタ、レジスタ、プログラムステータスワードを示すコンテキスト、個人情報等が格納される。

【0006】マルチタスク処理を行うためのタスクの状態としては、実行状態 (RUN)、実行可能状態 (READY)、待ち状態 (WAIT)、休止状態 (DORMANT) の4状態がある。実行状態は、カーネルによって実行権が渡されたタスクの状態である。待ち状態は、カーネルから実行権が渡されれば実行状態になれるタスクの状態である。待ち状態は、タスクが何らかの条件が満たされるのを待っている状態であり、実行条件が整えば実行可能となる。休止状態は、タスクが起動されていない状態又は終了後の状態である。

【0007】従来のリアルタイムオペレーティングシステムでは、アプリケーションのタスクから処理要求があると、タスクから処理要求を行い、リアルタイムオペレーティングシステムの該当する処理要求の中でコンテキスト等の退避を行い、処理要求の処理を行い、実行可能状態のタスクから次に実行するタスクをサーチし (スケジューリング)、実際に実行するためにコンテキスト等を復帰させ実行権を与える (ディスパッチ) ような処理が行われる。スケジューリングを行う場合には、タスクが持っているタスク優先度が考慮される。つまり、2つのタスクA、Bが実行可能状態で、先にタスクBの方が実行可能状態になったとしても、タスクAの方が高い優先度が与えられていたなら、タスクAに実行権が与えられる。

【0008】

【発明が解決しようとする課題】リアルタイムオペレーティングシステムでは、システムコールに対する応答時間を短縮することが重要である。システムコールに対する応答時間は、アプリケーションからシステムコールを発行して、次のアプリケーションに制御が戻ってくる時間であり、これはディスパッチ時間と呼ばれる。つまり、数々の事象に対して最適なのは実時間に応答することであるが、システムコールには、オーバーヘッドすなわちディスパッチ時間がある。したがって、この時間が如何に短縮されているかにより、リアルタイムオペレーティングシステムの性能を評価することができる。

【0009】そこで、各タスクに対応してレジスタバンクを用意することで、ディスパッチ時間を短縮することが考えられる。すなわち、従来では、システムコールが発行されると、コンテキストの退避/復帰の処理が行われる。このため、ディスパッチ時間が長くなる。これに対して、各レジスタバンクを各タスクに対応して用意しておけば、コンテキスト等の退避/復帰がレジスタバンクを切り替えることにより行なえる。

【0010】しかしながら、レジスタバンクの数は有限

である。このため、各タスクを各レジスタバンクに1対1に対応させると、並行して動作するタスクの数は、レジスタバンクの数に制限され、レジスタバンクの空きがなければ、処理を実行することができないことになる。このことは、アプリケーションを作成する上で制限事項となる。

【0011】したがって、この発明の目的は、並行して実行できるタスクの数が制限されず、ディスパッチ時間を短縮できるマルチタスク処理装置を提供することにある。

【0012】

【課題を解決するための手段】この発明は、複数のタスクが並列的に実行されるマルチタスク処理を行うマルチタスク処理装置において、各タスクに対応して占有される複数のレジスタバンクを有し、複数のレジスタバンクが各タスクに動的に割り当てられるようにしたマルチタスク処理装置である。

【0013】この発明では、更に、バンクの空き待ち行列手段を有し、各レジスタバンクは、そのレジスタバンクのタスクの処理が終了した時点でそのタスクを解放し、タスクを解放したバンクに空き待ち行列手段に待機されているタスクを割り当てるようにしている。

【0014】

【作用】複数のタスクに対応してレジスタバンクが用意される。このため、コンテキスト等の退避/復帰がレジスタバンクを切り替えることにより行なえる。そして、複数あるレジスタバンクが個々にオペレーティングシステムにより管理され、起動時に空いているレジスタバンクにタスクが動的に割り当てられ、タスクが終了されると、占有されていたレジスタバンクが解放される。オペレーティングシステムが管理しているレジスタバンクに空きが無ければ、起動したタスクはレジスタバンクの空き待ちの状態となり、タスクが占有しているレジスタバンクを解放するまで、待機される。このため、実行されるタスクの数がレジスタバンクの数に制限されなくなる。

【0015】

【実施例】以下、この発明の一実施例について図面を参照して説明する。図1は、この発明を実現するためのハードウェアの構成を示すものである。図1に示すように、この発明が適用された中央処理装置1は、CPUコア部2と、ROM3と、RAM4と、レジスタバンク5-1~5-nと、割り込みコントローラ6と、I/Oポート7とを含んでいる。レジスタバンク5-1~5-nは、例えばバンク数が15 ($n=15$)であり、これらのレジスタバンクは各タスクに占有される。このレジスタバンク5-1~5-nを利用することで、ディスパッチ時間が短縮できる。

【0016】すなわち、従来では、システムコールが発行されると、コンテキストの退避/復帰の処理が行われ

る。このため、ディスパッチ時間が長くなる。これに対して、各レジスタバンク5-1~5-nを各タスクに対応して用意しておけば、コンテキスト等の退避/復帰がレジスタバンク5-1~5-nを切り替えることにより行なえる。

【0017】しかしながら、レジスタバンク5-1~5-nの数は有限(例えば15)である。このため、各タスクを各レジスタバンク5-1~5-nに1対1に対応させると、並行して動作するタスクの数は、レジスタバンク5-1~5-nの数の数に制限され、レジスタバンク5-1~5-nの空きがなければ、処理を実行することができないことになる。

【0018】そこで、このようなレジスタバンク5-1~5-nの数によるタスクの数の制限を無くすために、複数あるレジスタバンク5-1~5-nが個々にオペレーティングシステムにより管理され、起動時に空いているレジスタバンク5-1~5-nにタスクが動的に割り当てられる。割り当てられたレジスタバンク5-1~5-nがそのタスク用の資源として占有される。タスクが終了されると、占有されていたレジスタバンク5-1~5-nが解放される。オペレーティングシステムが管理しているレジスタバンク5-1~5-nに空きが無ければ、起動したタスクはレジスタバンクの空き待ちの状態となり、タスクが占有しているレジスタバンク5-1~5-nを解放するまで、待機される。

【0019】以下、具体例について説明する。図2において、B1~B4は、レジスタバンクである。これらのレジスタバンクB1~B4は、図1におけるレジスタバンク5-1~5-nに対応している。レジスタバンク5-1~5-nのバンク数は例えば15であるが、ここでは説明を簡単とするため、バンク数を4としている。これらレジスタバンクB1~B4のうち、レジスタバンクB1~B3は実行される各タスクに割り当てられ、レジスタバンクB4はオペレーティングシステムOS用に割り当てられる。T1~T4はタスクであり、これらのタスクT1~T4は、同じ優先度のものである。Q1は、実行されるタスクが待機される実行可能タスク行列である。Q2は、レジスタバンク待ちのタスクが待機されるバンク空き待ち行列である。

【0020】4つのタスクT1~T4は、まず、実行可能タスク行列Q1に待機される。これらのタスクのうち、まず、タスクT1、T2の順に起動されるとすると、タスクT1にはレジスタバンクB1が資源として割り当てられ、タスクT2にはレジスタバンクB2が資源として割り当てられる。各レジスタバンクB1、B2は、処理が終了するまで、これらの各タスクT1、T2に占有される。

【0021】次に、タスクT3、T4が起動されるとする。この時、レジスタバンクB3は空いている。このため、タスクT3にレジスタバンクB3が資源として割り

当てられる。これで、全てのレジスタバンクB1～B4が使われたことになり、タスクT4に割り当てべきレジスタバンクがなくなる。そこで、タスクT4は、バンク空き待ち行列Q2に待機される。

【0022】その後、レジスタバンクB1～B3に割り当てられたいずれかのタスクの処理が終了して休止状態になったとすると、オペレーティングシステムOSは、バンク待ち行列Q2に待機されていたタスクT4を、処理の終了したタスクのレジスタバンクに割り当てる。

【0023】図3及び図4は、第2の例である。この例では、5つのタスクT11、T12、T13、T14、T15が実行可能タスク行列Q1に待機される。図3に示すように、これらのタスクのうち、タスクT13、T14、T15の順に起動されるとすると、タスクT13にはレジスタバンクB1が資源として割り当てられ、タスクT14にはレジスタバンクB2が資源として割り当てられ、タスクT15にはレジスタバンクB3が資源として割り当てられる。各レジスタバンクB1～B3は、処理が終了するまで、これらの各タスクT13、T14、T15に占有される。

【0024】次に、タスクT11、T12が起動されるとする。この時、レジスタバンクB1、B2、B3は、タスクT13、T14、T15に占有されているため、タスクT11、T12に割り当てべきレジスタバンクがない。このため、タスクT11、T12は、バンク空き待ち行列Q2に待機される。

【0025】その後、タスクT14が処理を終了して休止状態になったとすると、図4で破線で示すように、オペレーティングシステムOSは、タスクT14に占有されていたレジスタバンクB2を解放し、バンク待ち行列Q2に待機されていたタスク（この場合にはタスクT11）にこのレジスタバンクB4を割り当てる。更にその＊

＊後、レジスタバンクB1～B3に割り当てられたいずれかのタスクの処理が終了して休止状態になったときに、オペレーティングシステムOSは、バンク待ち行列Q2に待機されていたタスクT12を、処理の終了したタスクのレジスタバンクに割り当てる。

【0026】

【発明の効果】この発明によれば、複数のタスクに対応してレジスタバンクが用意される。このため、コンテキスト等の退避／復帰がレジスタバンクの切り替えることにより行なえ、ディスパッチ時間が短縮できる。そして、複数あるレジスタバンクが個々にオペレーティングシステムにより管理され、起動時に空いているレジスタバンクにタスクが動的に割り当てられ、タスクが終了されると、占有されていたレジスタバンクが解放される。オペレーティングシステムが管理しているレジスタバンクに空きが無ければ、起動したタスクはレジスタバンクの空き待ちの状態となり、タスクが占有しているレジスタバンクを解放するまで、他のタスクが待機される。このため、実行されるタスクの数がレジスタバンクの数に制限されなくなる。

【図面の簡単な説明】

【図1】この発明が適用できる中央処理装置の一例のブロック図である。

【図2】この発明の一実施例の動作説明図である。

【図3】この発明の一実施例の動作説明図である。

【図4】この発明の一実施例の動作説明図である。

【符号の説明】

5-1～5-n, B1～B4 レジスタバンク

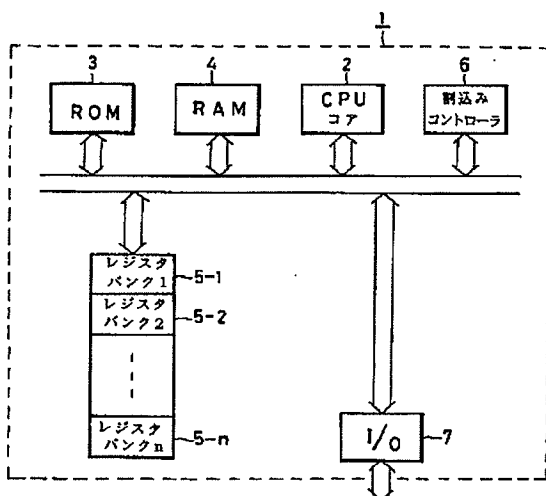
T1～T4, T1～T15 タスク

OS オペレーティングシステム

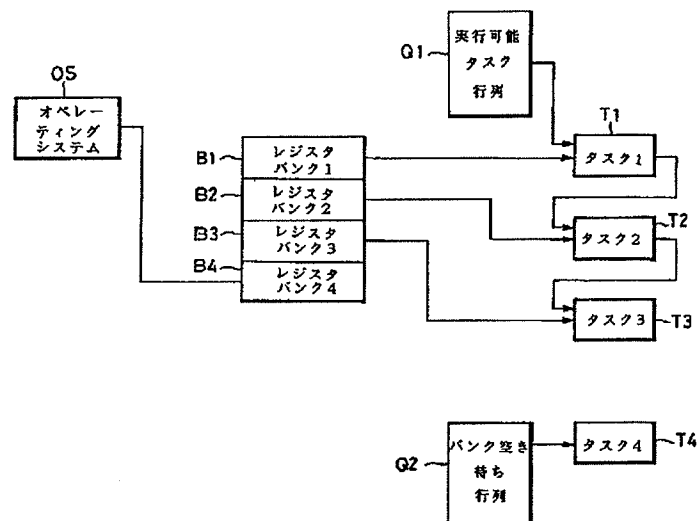
Q1 実行可能タスク行列

Q2 バンク空き待ち行列

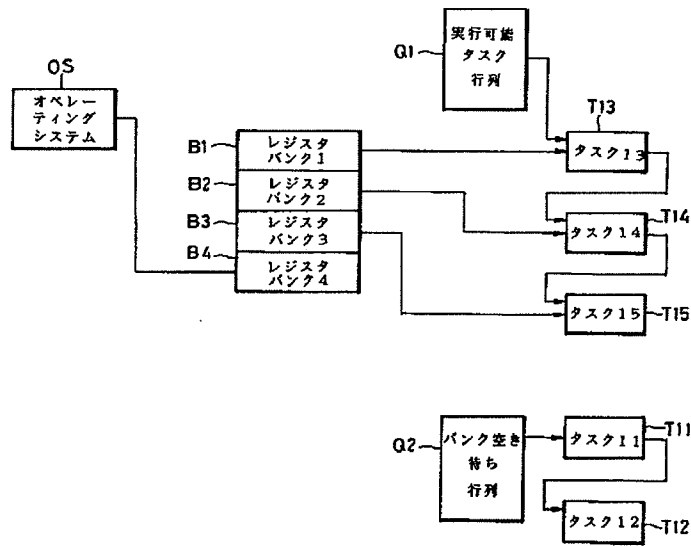
【図1】



【図2】



【図 3】



【図 4】

